KLAUSUR
zur Lehrveranstaltung "Algorithmen"

Name: [redacted] [redacted]

[redacted]-Punkte: –.......

4.4.08

Aufgabe 1 (6 Punkte):

6

Ein schwach zusammenhängender, schlichter Digraph **D = (V,A)** mit **V =** $\{v_1, \ldots, v_{14}\}$ sei durch folgende Adjazenzlisten gegeben:

A_1=[4,6,10,11], A_2=[7,8,9,12,14], A_3=[1,5,10,11,12], A_4=[5,6,11,13],
A_5=[1,2,8,10], A_6=[3,7,10,13], A_7=[1,3,4,13], A_8=[3,7,9,12],
A_9=[3,6,7,14], A_{10}=[2,4,12,14], A_{11}=[2,5,6,13], A_{12}=[4,5,9,14],
A_{13}=[1,8,9,10], A_{14}=[2,5,8,11].

Besitzt **D** einen gerichteten Euler-Weg? $\boxed{\text{NEIN}}$

Begründen Sie Ihre Antwort!

-
-

~~Ae~~	d-	d+
A_1	4	4
A_2	5	4
A_3	5	4
A_4		
A_5		
A_6		
A_7		
A_8		
A_9		
A_{10}		
A_{11}		
A_{12}		
A_{13}		
A_{14}		

Es gibt zwei Knoten die mehr ausgehende Kanten besitzen als eingehende!

Aufgabe 2 (6 Punkte):

4

Ordnen Sie die folgenden Funktionen $f_1, \ldots f_5$ aufsteigend nach ihrer Laufzeit-Komplexität, und geben Sie die Reihenfolge in der Form $O(f_i) \subset O(f_k) \subset \ldots$ an!

$f_1 = 5000 \log n$ $\quad f_2 = 75n^3 + 3n$ $\quad f_3 = 10^{50}$ $\quad f_4 = \frac{4n^2}{(n+1)}$ $\quad f_5 = n + 2n \log n$

$O(n) \subset O(n^3)$

$O(f_3) \subset O(f_1) \subset O(f_5) \subset O(f_4) \not\subset O(f_2)$

Aufgabe 3 (7 Punkte):

Ein Graph **G = (V,E)** mit **V =** $\{v_1, \ldots, v_{14}\}$ sei durch folgende Adjazenzlisten gegeben: 0

A_1=[3,4,5,6,7,8,9,10,11,12,13,14], A_2=[3,4],

A_3=[1,2,4,5,6,8], A_4=[1,2,3,5,7,9], A_5=[1,3,4,6,7,10],

A_6=[1,3,5,8,9,11], A_7=[1,4,5,8,9,12], A_8=[1,3,6,7,9,13],

A_9=[1,4,6,7,8,14], A_{10}=[1,5,11,12,13,14], A_{11}=[1,6,10,12,13,14],

A_{12}=[1,7,10,11,13,14], A_{13}=[1,8,10,11,12,14], A_{14}=[1,9,10,11,12,13].

a) Ermitteln Sie die Hülle von **G**: **c(G) =** K_{14} f

Erläutern Sie kurz Ihre Lösung!

~~Es gibt 14 Knoten~~

f Es gibt 14 Knoten, also ist ~~[illegible]~~

$c(G) = K_{14}$

b) Ist aus Ihrem Ergebnis zu **a)** mittels der in der Vorlesung gelernten Sätze ableitbar, ob **G** ein Hamilton-Graph ist? Nein ! Widerspruch zu a) !

Aufgabe 4 (4 Punkte): 4

Gegeben ist der Graph **G** = (V,E) in Bild 1 mit dem Matching

M = { [a,c], [b,d] }.

Geben Sie einen **M**-erweiternden Weg **W** in G an!

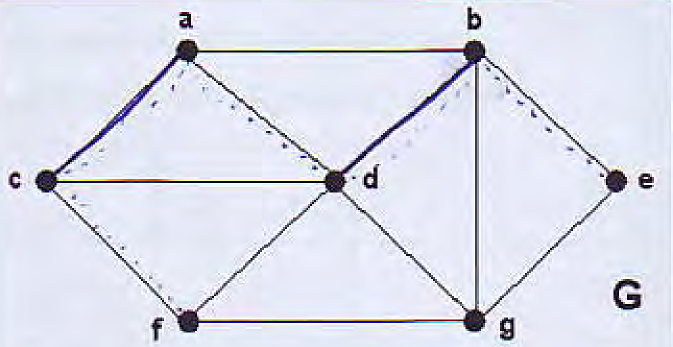

Bild 1

W = {fc, ca, ad, db, be}

15

Aufgabe 5 (15 Punkte):

Ein schlichter Graph **G** mit 30 Knoten $u_1, \ldots, u_{30}$ sei durch folgende Adjazenzlisten gegeben :

A_1=[2,17], A_2=[1,16,28], A_3=[5,19], A_4=[5,6,19],

A_5=[3,4,6,12,29], A_6=[4,5,7,29], A_7=[6,21], A_8=[11,22],

A_9=[13,26,29], A_{10}=[13,14,15,18], A_{11}=[8,15,20,21,23], A_{12}=[5,19,24],

A_{13}=[9,10,18,26,29], A_{14}=[10,18,25], A_{15}=[10,11,20,25], A_{16}=[2,28],

A_{17}=[1,24], A_{18}=[10,13,14], A_{19}=[3,4,12], A_{20}=[11,15,23],

A_{21}=[7,11,25,27], A_{22}=[8,23], A_{23}=[11,20,22], A_{24}=[12,17,30],

A_{25}=[14,15,21,29], A_{26}=[9,13,27], A_{27}=[21,26], A_{28}=[2,16,30],

A_{29}=[5,6,9,13,25], A_{30}=[24,28].

Zur Bestimmung eines kürzesten Weges von u_1 zu allen anderen Knoten von **G** soll das BFS-Programm von Moore (Skript, S. 42) verwendet werden.

Dokumentieren Sie den Beginn des Programmablaufs, indem Sie in die folgende Tabelle die ersten 12 Elemente in der Reihenfolge eintragen, in der sie in die Warteschlange **Q** hinein gelangen, und die Markierungen **m**, die die betreffenden Knoten im Laufe des Algorithmus bekommen!

Beachten Sie dabei die in den Adjazenzlisten vorhandene Reihenfolge!

Q	1	2	17	16	28	24	30	12	5	19	3	4
m	0	1	1	2	2	2	3	3	4	4	5	5

Ermitteln Sie auf der Grundlage dieses Ergebnisses mittels des rückverfolgenden Algorithmus (Skript, S. 42) den kürzesten Weg vom Knoten u_1 zum Knoten u_3!

Beachten Sie dabei die in den Adjazenzlisten vorhandene Reihenfolge!

Dokumentieren Sie den Ablauf des Algorithmus in der folgenden Tabelle!

i	5	4	3	2	1	STOP	
V_i	3	5	12	24	17	1	

Fassen Sie den in der Tabelle ermittelten kürzesten Weg zusammen :

W = $u_1 \; u_{17} \; u_{24} \; u_{12} \; u_5 \; u_3$

Aufgabe 6 (15 Punkte): 13

Das öffentliche Verkehrsnetz der Stadt **Algotal** ermöglicht folgende Direkt-Fahrzeiten (in Minuten) zwischen den wichtigsten Verkehrsknotenpunkten:

	Amselplatz	**B**adstraße	**C**harlottenhof	**D**arwinallee	**E**insteinufer	**F**riedenstraße
Amselplatz	-	40	12	75	45	60
Badstraße	40	-	50	20	25	30
Charlottenhof	12	50	-	58	30	20
Darwinallee	75	20	58	-	12	32
Einsteinufer	45	25	30	12	-	15
Friedenstraße	60	30	20	32	15	-

Ermitteln Sie mittels des Algorithmus von Dijkstra die kürzesten Fahrzeiten vom **A**mselplatz zu jedem anderen Knotenpunkt (reine Fahrzeit, ohne Wartezeiten beim Umsteigen)!
Dokumentieren Sie den Ablauf des Algorithmus in der folgenden Tabelle!

M O	0 A	∞ B	∞ C	∞ D	∞ E	∞ F
M O	0	40 B	12 C	75 D	45 E	60 F
M O		~~40~~ 40 B	~~12~~ 12 ~~C~~	~~60~~ 70 D	~~45~~ 42 E	~~60~~ 32 ! F
M O	0	40	12	60 Folgefehler D	42 E	32 F
M O	0	40	12	60 D	42 E	32
M O	0	40	12	54 D	42	32
M O	0	40	12	54	42	32

Tragen Sie die ermittelten kürzesten Fahrzeiten in die folgende Tabelle ein:

Vom **A**mselplatz nach	**B**adstraße	**C**harlottenhof	**D**arwinallee	**E**insteinufer	**F**riedenstraße
Kürzeste Fahrzeit	40	12	54	42	32

Aufgabe 7 (13 Punkte):

13

Ein schlichter, orientierbarer Graph **G=(V,E)** mit 35 Knoten $v_1, \dots, v_{35}$ sei durch folgende Adjazenzlisten gegeben :

A_1=[5,12,17,22,30],	A_2=[3,23,24],	A_3=[2,14],	A_4=[22,23],	A_5=[1,12,19,25],
A_6=[24,32],	A_7=[24,28],	A_8=[20,26],	A_9=[27,28,33],	A_{10}=[23,32,35],
A_{11}=[18,34],	A_{12}=[1,5,25],	A_{13}=[27,29],	A_{14}=[3,29,31],	A_{15}=[16,26],
A_{16}=[15,33],	A_{17}=[1,22,23],	A_{18}=[11,35],	A_{19}=[5,30],	A_{20}=[8,31],
A_{21}=[29,33],	A_{22}=[1,4,17],	A_{23}=[2,4,10,17],	A_{24}=[2,6,7,29],	A_{25}=[5,12],
A_{26}=[8,15],	A_{27}=[9,13],	A_{28}=[7,9],	A_{29}=[13,14,21,24],	A_{30}=[1,19],
A_{31}=[14,20],	A_{32}=[6,10,34],	A_{33}=[9,16,21],	A_{34}=[11,32],	A_{35}=[10,18].

Mit Hilfe des Algorithmus von Hopcroft und Tarjan soll eine stark zusammenhängende Orientierung für **G** ermittelt werden.

a) Dokumentieren Sie in der folgenden Tabelle den Beginn des Algorithmus (d.h. nur die vorhandenen Tabellenfelder sollen ausgefüllt werden)! Beginnen Sie beim Knoten v_1 und wählen Sie im Falle mehrerer Möglichkeiten immer den Knoten mit dem kleineren Index!

Knoten- Markierung	Kanten-Orientierung	
$m(v_1) = 1$		~~v_1~~
$m(v_5) = 2$	$v_1 \to v_5$	~~$v_1 v_5$~~
$m(v_{12}) = 3$	$v_5 \to v_{12}$	~~$v_1 v_5 v_{12}$~~
$m(v_{25}) = 4$	$v_{12} \to v_{25}$	~~$v_1 v_5 v_{12} v_{25}$~~
$m(v_{19}) = 3$	$v_5 \to v_{19}$	~~$v_1 v_5 v_{12} v_{25} v_{19}$~~
$m(v_{30}) = 4$	$v_{19} \to v_{30}$	~~$v_1 v_5 v_{12} v_{25} v_{19}$~~ v_{30}
$m(v_{17}) = 2$	$v_1 \to v_{17}$	~~$v_1 v_5 v_{12} v_{25} v_{19} v_{30}$~~ v_{17}

b) Welche Orientierung bekommt die Kante $[v_5, v_{25}]$ in Schritt (4) des Algorithmus (Skript Seite 66)? Verbinden Sie die beiden Knoten durch den entsprechenden Pfeil!

$$v_5 \longleftarrow v_{25}$$

Aufgabe 8 (5 Punkte): 5

Gegeben ist das Netzwerk **N** = (V,A,c) in Bild 2 mit der Quelle **s**, der Senke **t** und den eingetragenen Kapazitäten.

Ermitteln Sie die Kapazität des Schnittes A(S,T) mit S = {s,a,c,e}, und geben Sie dabei Ihren Rechenweg mit an!

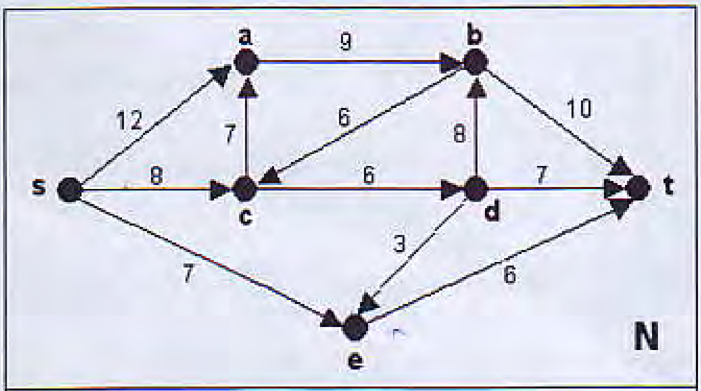

Bild 2

$S = \{s, a, c, e\}$

$T = \{b, d, t\}$

~~A(S,T) = { ... sa,~~

$A(S,T) = \{ab, cd, et\}$

$c(S,T) = 9 + 6 + 6 = \underline{\underline{21}}$

Aufgabe 9 (4 Punkte): 3

Auf die folgende Eingabefolge soll der <u>HeapSort-Algorithmus</u> angewendet werden, um sie aufsteigend zu sortieren.

24	29	16	20	33	11	7	18

Notieren Sie die Eingabefolge nach der ersten Vertauschung von 2 Elementen, beschreiben Sie kurz den Ablauf des Algorithmus bis zu dieser Vertauschung und begründen Sie, weshalb diese Vertauschung stattfindet!

~~24~~	~~29~~	~~16~~	20	29	11	7	18
24	33	16					

zu ungenau

Es musste vorher nichts vertauscht werden da jedes mal die Max-Heap-Eigenschaft nicht verletzt wurde. Erst als die 29 mit 33 verglichen wurden musste eine vertausch stattfinden, damit die 33 nach oben rücken konnte, und somit ist die Max-Heap-Eigenschaft erst einmal wieder gewährt.